



# Deep transfer learning

- Domain Adaptation, Transfer learning
- Style Transfer, Multi-view learning

# Pourquoi ?

- Deep learning aboutit à des performances inégalées en vision, speech, ...
- Mais nécessite de nombreux exemples pour l'entraînement, pas toujours disponibles
  - apprendre avec peu de données ?
- Ne généralise pas toujours bien sur de nouvelles données si trop différentes de celles vues en entraînement
  - $P(Y_s) \neq P(Y_t)$  ou  $P(X_s) \neq P(X_t)$  ?



What your net is trained on



What it's asked to label

**“Dataset Bias”**  
**“Domain Shift”**  
**“Domain Adaptation”**  
**“Domain Transfer”**

Source: <http://www.eecs.umich.edu/eecs/pdfs/events/4142.pdf>

# Pourquoi ?

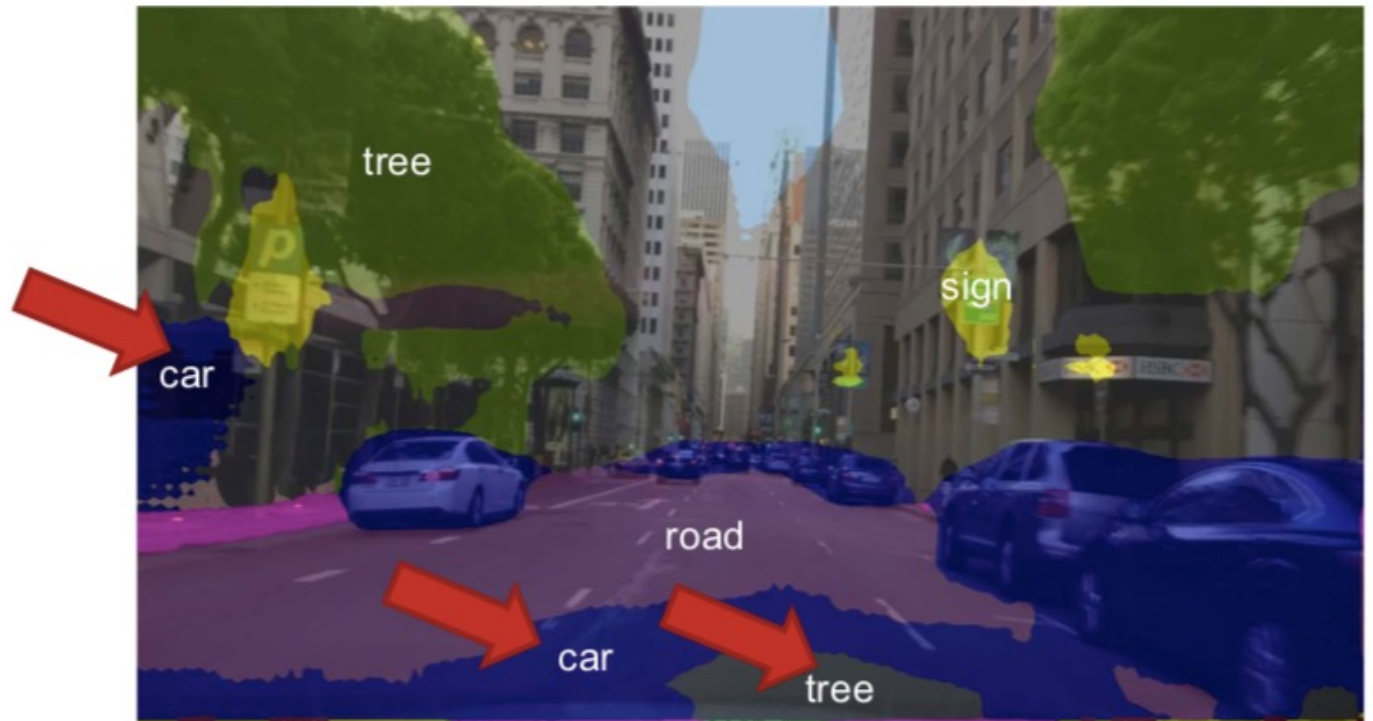
- Image segmentation
- Entrainement et Test sur le même dataset (Cityscapes)



Source: <http://www.eecs.umich.edu/eecs/pdfs/events/4142.pdf>

# Pourquoi ?

- Image segmentation
- Entrainement et Test sur différents datasets (Cityscapes vs SF)



Source: <http://www.eecs.umich.edu/eecs/pdfs/events/4142.pdf>

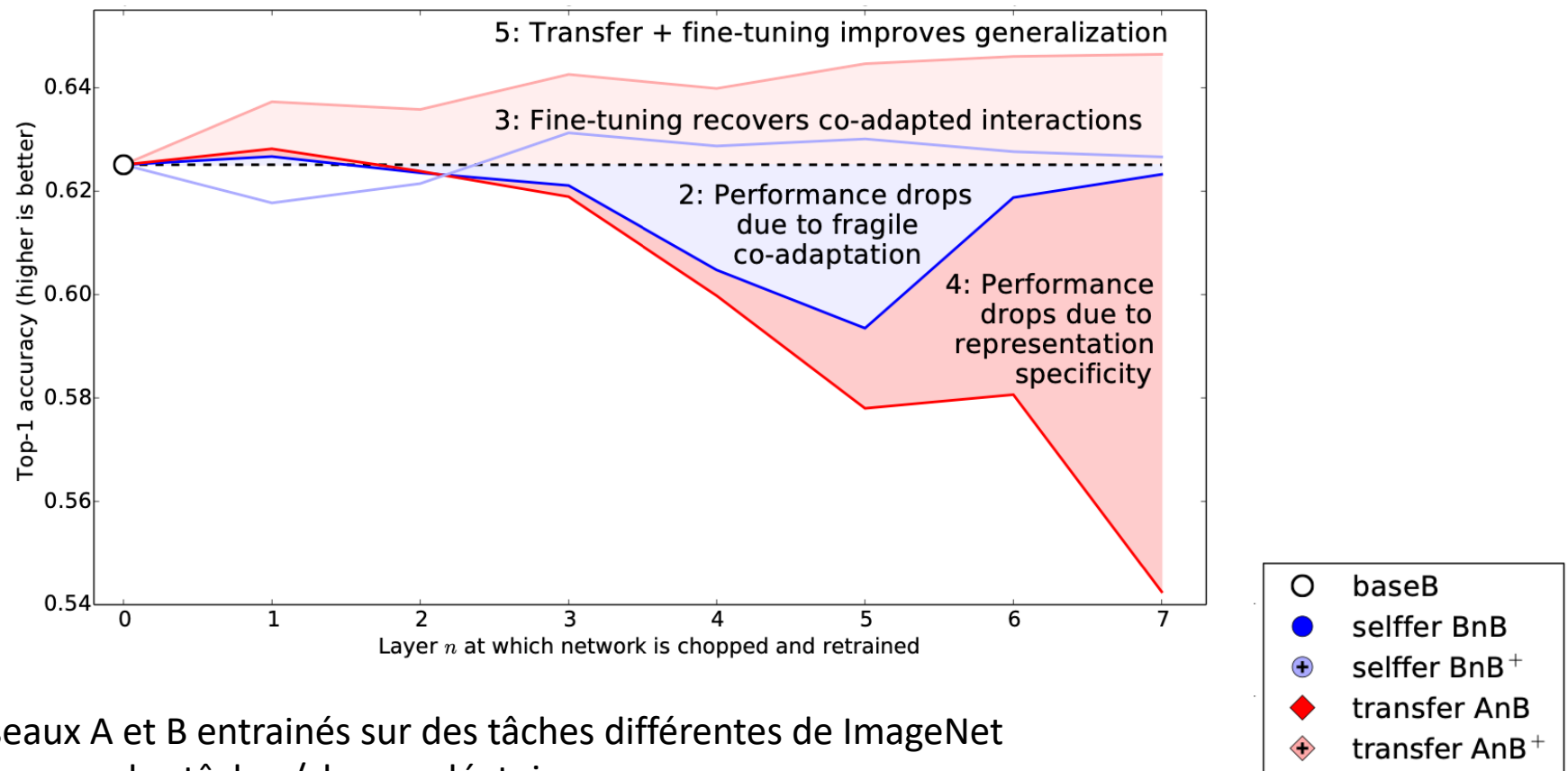
# Transfert Learning - Finetuning

- Peut-on transférer des connaissances apprises sur un domaine et les utiliser sur un autre domaine ?
  - Exemple : réutiliser des filtres de convolutions appris sur ImageNet
- Idée simple : Finetuning
  - Plutôt que d'initialiser les couches de convolutions aléatoirement, initialiser avec les poids provenant d'un modèle déjà entraîné
  - Permet (ou non, ou peu) de modifier les poids par descente de gradient sur une nouvelle tâche
  - Utile mais pas toujours..

# Généricité des représentations [Yosinsky et al., 2014]

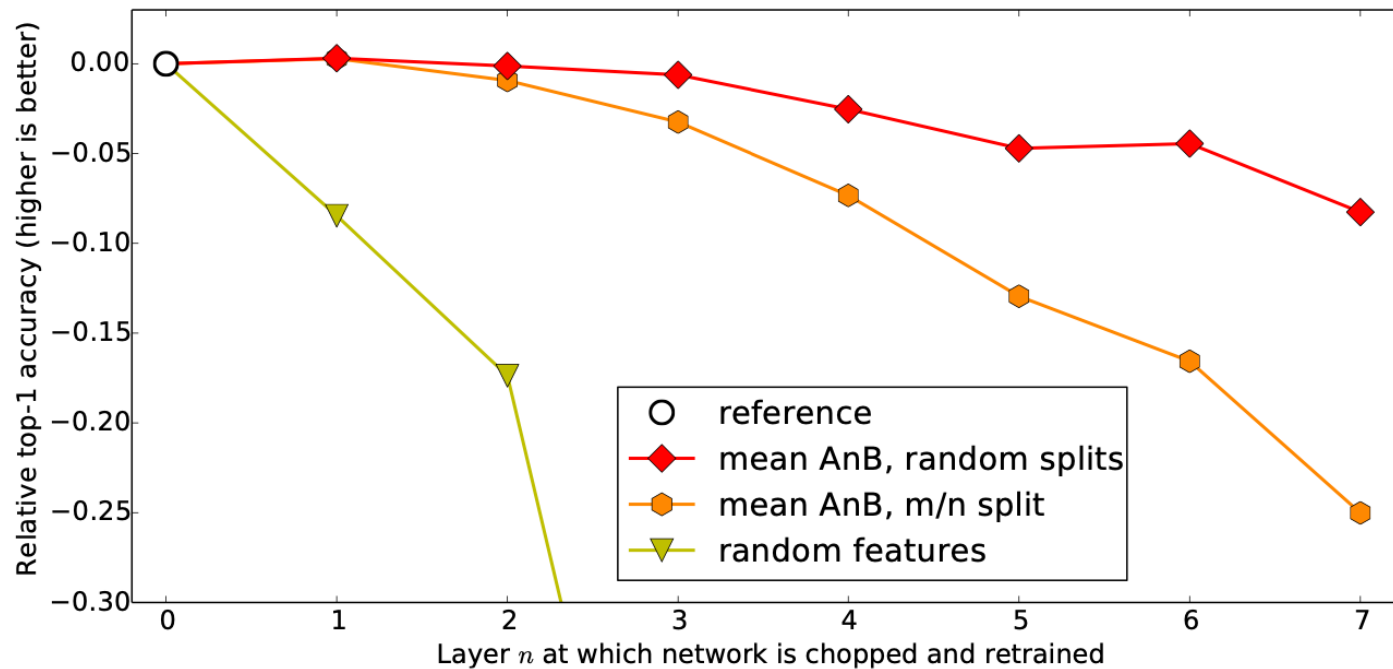
- Expériences sur deux tâches similaires
  - 2 réseaux : Vert appris sur A – Bleu appris sur B
  - Utiliser Vert sur B (et vice versa) avec ou sans finetuning
  - Etude de l'effet du finetuning jusqu'à la couche  $i$
- Les couches basses sont plus transférables que les plus hautes

# Généricité des représentations [Yosinsky et al., 2014]





# Généricité des représentations [Yosinsky et al., 2014]

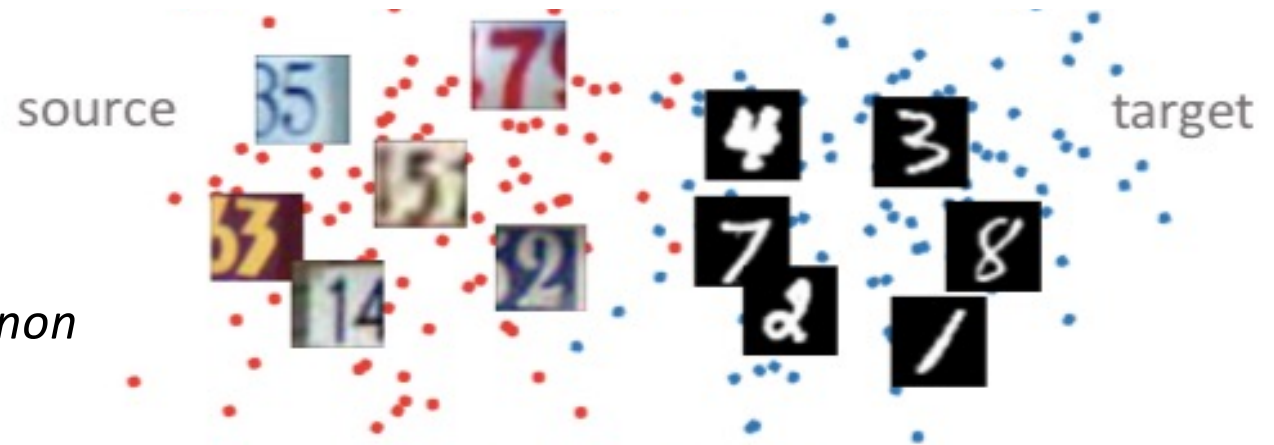


- Le transfert se dégrade pour des tâches plus distantes
- Mieux que le hasard

# Adaptation de domaine

- Classification de données provenant de distributions différentes

- *Source vs Target*
- Classes similaires
- Few-shot learning
- *Etiquettes cibles connues ou non*



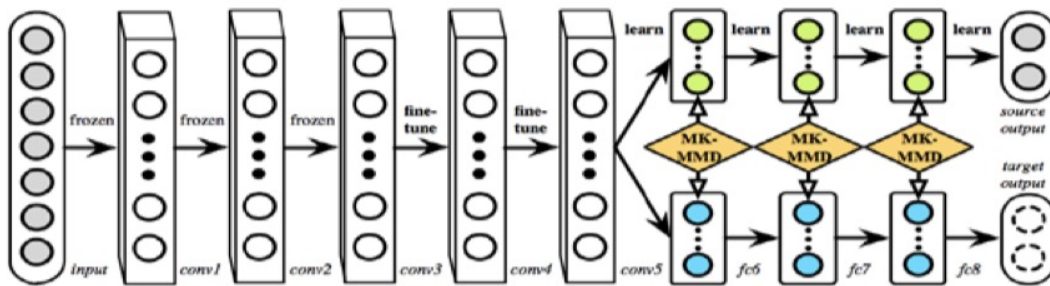
Source Domain  $\sim P_S(X, Y)$   
lots of **labeled** data

$\neq$

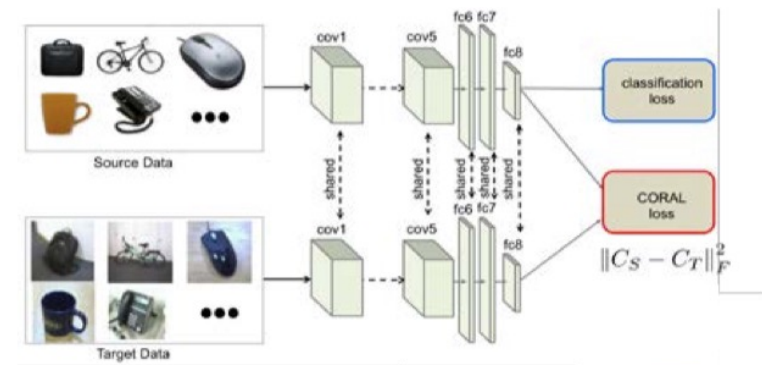
Target Domain  $\sim P_T(Z, H)$   
unlabeled or limited labels

# Adaptation de domaine

- Classification de données provenant de distributions différentes
  - *Source vs Target*
- Forcer l'alignement des domaines [Long et al., 2015] [Sun & Saenko, 2016]



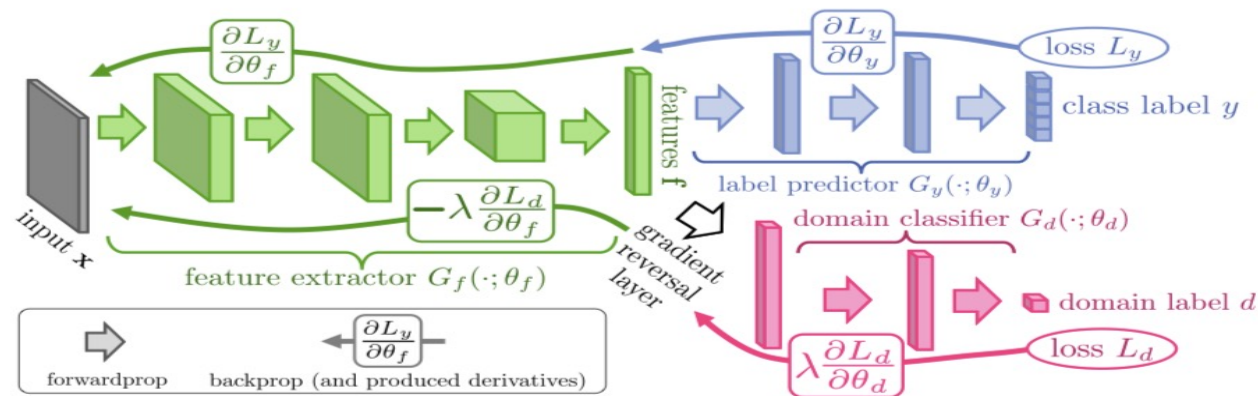
Maximum Mean Discrepancy [Long, 2015]



CORrelation ALignment [Sun & Saenko, 2016]

# Adaptation de domaine

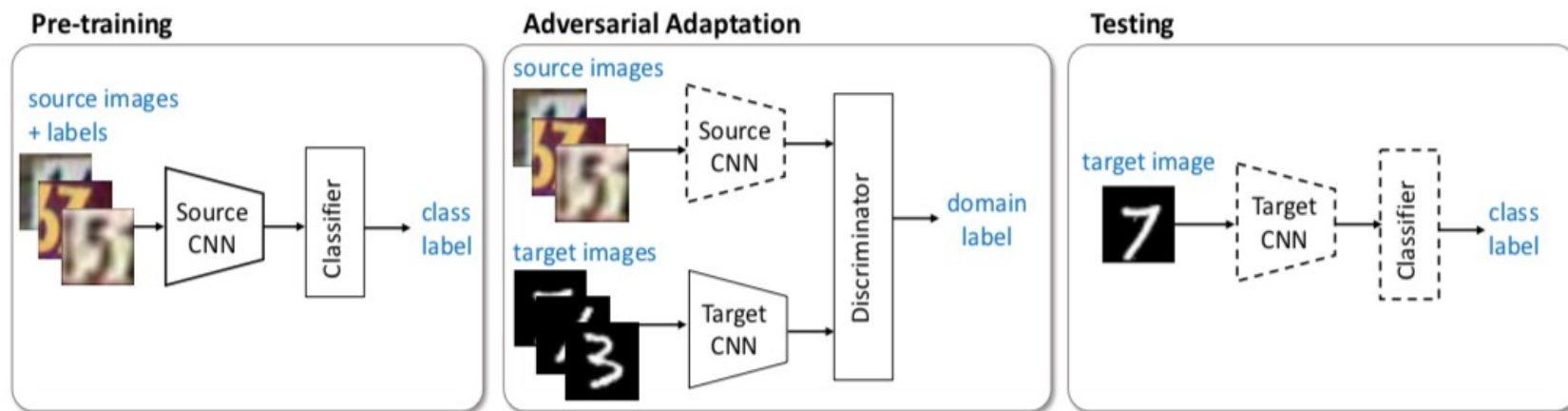
- Classification de données provenant de distributions différentes
  - *Source vs Target*
- Par adaptation « Adversarial » [Ganin, 2016] [Tseng, 2017]



Domain-Adversarial Training of Neural Networks [Ganin, 2016]

# Adaptation de domaine

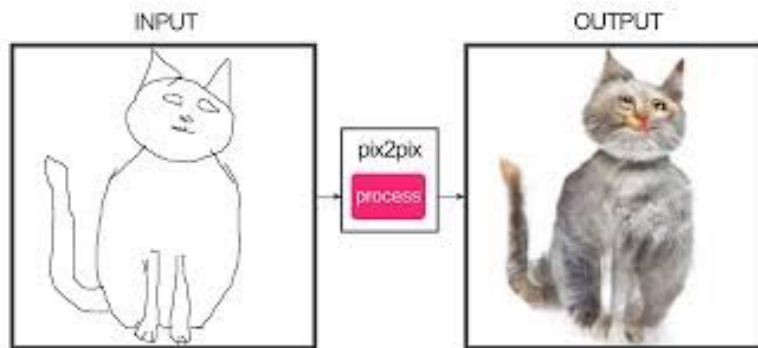
- Classification de données provenant de distributions différentes
  - *Source vs Target*
- Par adaptation « Adversarial » [Ganin, 2016] [Tseng, 2017]



Adversarial Discriminative Domain Adaptation [Tzeng, 2017]

# Transformation d'images

- Image-to-image translation with conditional adversarial networks [Isola, 2017] → « *pix2pix* »
  - L2 + CGAN
  - Discriminateur par patch



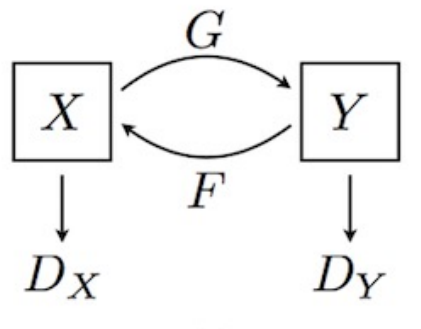
# Transformation d'images

- Image-to-image translation with conditional adversarial networks [Isola, 2017] → « *pix2pix* »

- L2 + CGAN

- Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks [Zhu, 2017]

→ CycleGAN :  $\mathcal{L}_{\text{cyc}}(G, F) = \mathbb{E}_{x \sim p_{\text{data}}(x)} [\|F(G(x)) - x\|_1] + \mathbb{E}_{y \sim p_{\text{data}}(y)} [\|G(F(y)) - y\|_1]$

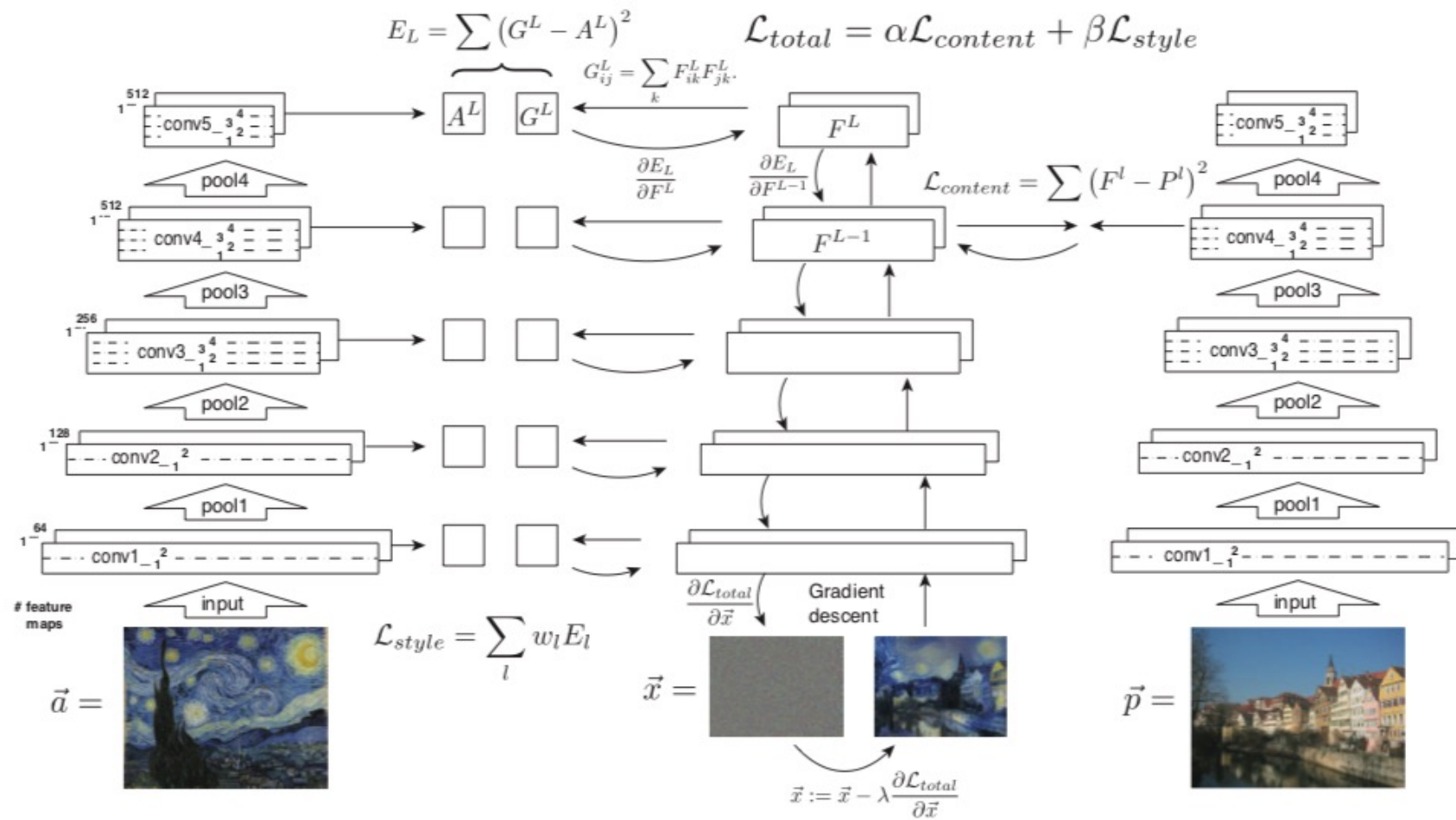


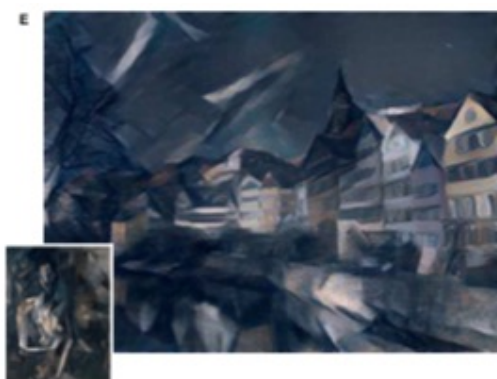
# Image Style Transfert

- Transférer le style d'une image vers une autre, dont on conserve le contenu
- Idée :
  - Utiliser un modèle convolutionnel préentraîné, et séparer explicitement le style  $S$  et le contenu  $C$ , à partir d'une paire d'images  $(C, S)$
  - Générer une nouvelle image dont le contenu correspond à  $C$  et le style correspond à  $S$
  - Ne nécessite pas réellement d'apprentissage, mais une mise à jour itérative de l'image générée à partir des gradients  $\nabla C$  et  $\nabla S \rightarrow$  utilise L-BFGS



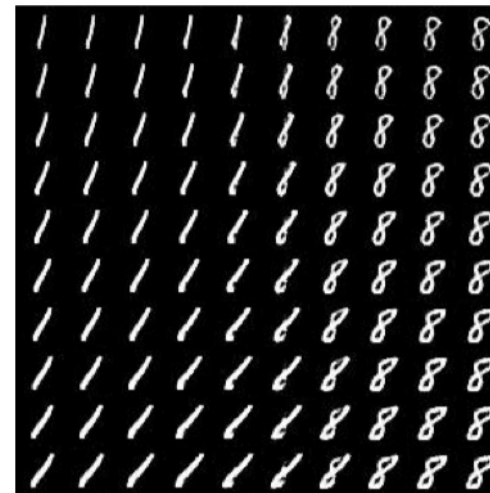
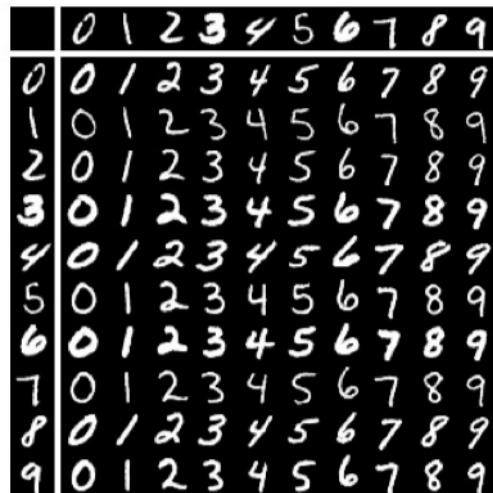
# Image Style Transfer Using CNN [Gatys et al., 2016]





# Apprendre à démêler contenu / style

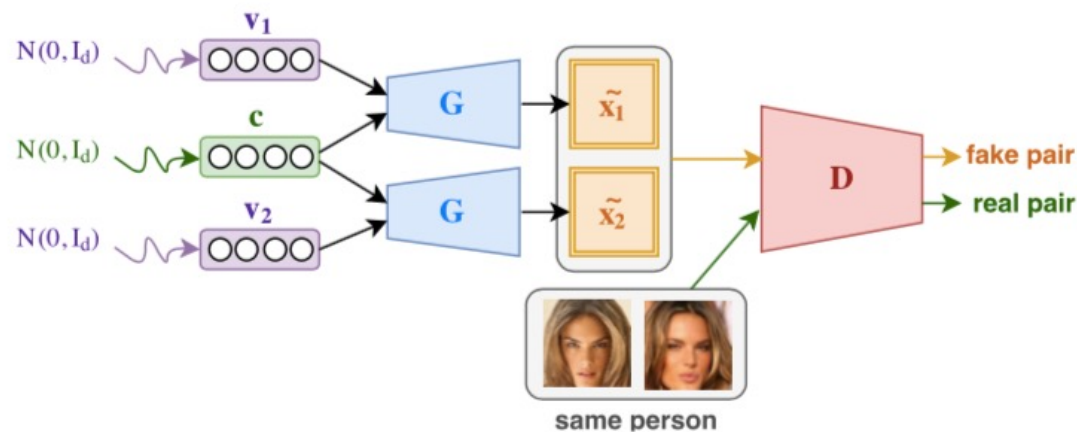
- « Disentangling »
  - Apprendre des représentations où les facteurs de variations sont associés à des neurones identifiables, ex: 1 neurone par facteur
  - Générer des images avec un autre style
  - Modèles conditionnels, Discriminateur avec pair d'images [Mathieu, 2016]



# Adversarial Learning for Generating Multiview Data

[Chen, 2018]

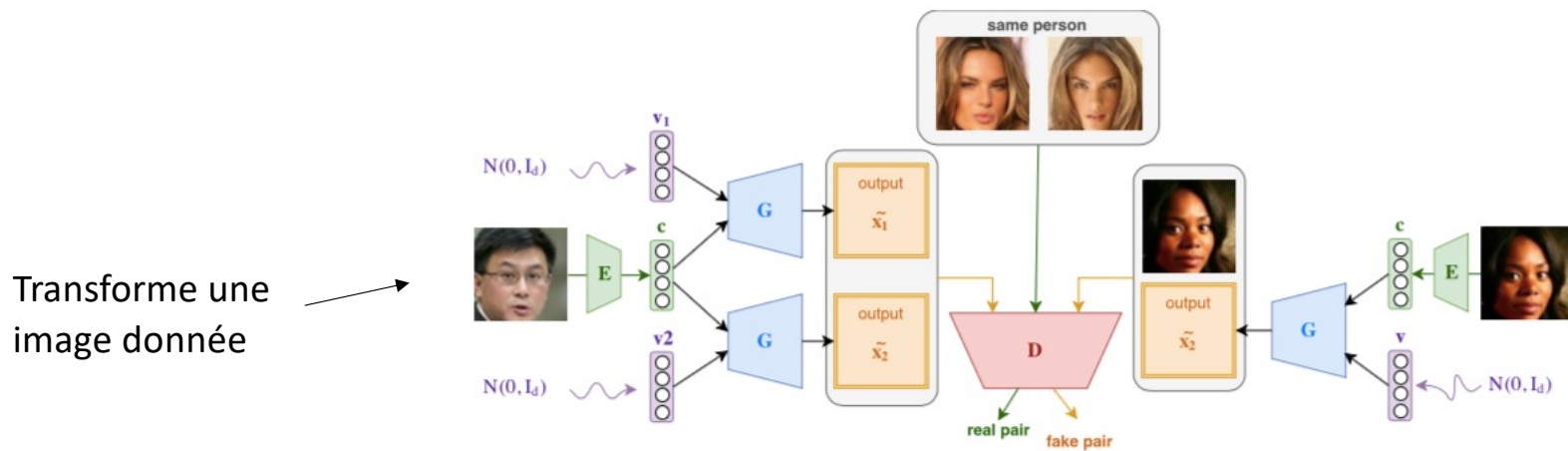
- Séparation contenu / vues : Génère des images avec une autre vue [Chen, 2018]
  - Deux (copies) du générateur produisent deux images avec le même  $c$  et un  $v$  différent
  - Le discriminateur apprend à distinguer une paire d'images d'une même personne dans deux vues quelconques



# Adversarial Learning for Generating Multiview Data

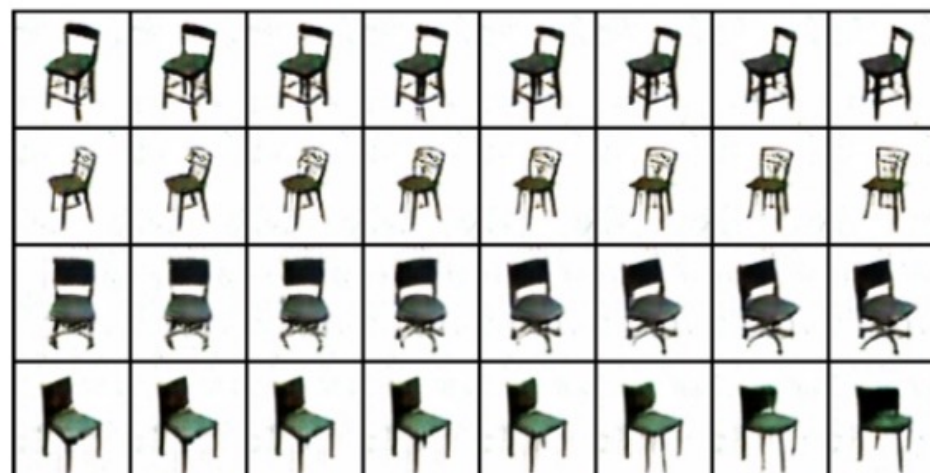
[Chen, 2018]

- Séparation contenu / vues : Génère des images avec une autre vue [Chen, 2018]
  - Deux (copies) du générateur produisent deux images avec le même  $c$  et un  $v$  différent
  - Le discriminateur apprend à distinguer une paire d'images d'une même personne dans deux vues quelconques





# Adversarial Learning for Generating Multiview



# Motion Capture Synthesis with Adversarial Learning [Qi, 2017]

- Même idée pour la génération de séquences

